

## REMARKS

In the Office Action the Examiner noted that claim 1-8 were pending in the application and the Examiner rejected all claims. By this Amendment, various claims have been amended and new claims 9-12 has been added. Thus, claims 1-12 are pending in the application. The Examiner's rejections are traversed below.

### The Specification

In items 3-5 of the Office Action the Examiner objected to the specification and title. The specification and title have been amended in accordance with the Examiner's comments and to generally improve its form. A substitute specification is attached. Therefore, it is submitted that the specification, as amended, should be acceptable.

### Claim Objections

Claim 2 has been amended in accordance with the Examiner's comment in item 6 on page 2 of the Office Action. It is submitted that claim 2 as amended is in appropriate form.

### Rejection Under 35 U.S.C. § 112

The claims have been amended in accordance with the Examiner's comments in item 7-9 on page 3 of the Office Action. It is submitted that the claims, as amended, meet the requirements of 35 U.S.C. § 112.

### Prior Art Rejections of Claims 1-2

In item 10 on pages 3-7 of the Office Action the Examiner rejected claims 1-2 under 35 U.S.C. § 103 as unpatentable over U.S. Patent 5,298,007 to Miyajima in view of U.S. Patent 6,088,624 to Khan et al. or U.S. Patent 5,640,559 to Silberbauer et al. In item 11 on pages 7-9 of the Office Action the Examiner rejected claims 1-2 as unpatentable over U.S. Patent 5,578,913 to Yasuda et al. in view in Kahn et al. or Silberbauer et al. Thus, the Examiner presents two separate rejections which employ either Yasuda et al. or Miyajima as the primary reference.

### The Prior Art

U.S. Patent 5,290,006 to Miyajima is directed to a machining method of a punch press machine by which a moving time of a workpiece table is excluded in a punch press machining to improve operation efficiency. Figure 2 is a block diagram of a numerical control machine and Figure 3 is a diagram explaining an example of a machining program which is stored in a CMOS 24 (Figure 2). The machining program illustrated in Figure 3 illustrates a macro command which begins with "U" and ends with "V" so that a machining command P2 is U90 and V90 of a machining command P6 is a macro end command.

U.S. Patent 5,578,913 to Yasuda et al. is directed to a numerical control device which can perform machining processes in a short time while maintaining high precision regardless of the order in which the programs are commanded. Consecutive blocks of a numerical control program are retrieved on a block-by-block basis, and a pre-execution end position and a post-execution end position value for each of the blocks of the numerical control device are set and stored in a ROM (see Abstract). A non-volatile memory 53 (Figure 2) stores parameters such as end-position values as illustrated in the Appendix at Table I (see column 13) post-execution end-position values and pre-execution end-position values for each axis are stored in the non-volatile memory 53 according to the combination of commands to be executed. ROM 51 stores programs for overall control of the numerical control device 50 and a ROM 52 stores data such as in-position values for before and after execution of each axis (column 4, lines 30-53).

U.S. Patent 6,088,624 to Kahn is directed to an industrial controller with flexible data structures. The Examiner cited the Kahn reference for its disclosure relating to length.

U.S. Patent 5,640,559 to Silberbauer et al. is directed to a method for encoding and decoding units of data representing the method for encoding and decoding stored units of data representing entity/relationship data, function calls and file data. The Examiner cited Silberbauer on the basis that it discloses encoding a length of the unit of data into a length field of a prefix for the data unit.

### The Present Claimed Invention Patentably Distinguishes Over the Prior Art

The present invention as set forth in claim 1 is directed to a numerical controller for controlling a machine according to a machine program. A storage device or medium stores input/output units each including program block data obtained by dividing the machining program among the input/output units. Each input/output unit includes additional information associated

with the program block data. The additional information includes an effective data length of the program block, front input /output unit data designating an input/output unit immediately preceding each input/output unit in a sequence of the machining program, and rear input/output unit data designating an input/output unit following each input/output unit in a sequence of the machining program. A processor processes the input/output units. An interface inputs/outputs the input/output units between the storage device or medium and the processor. It is submitted that neither Miyajima et al. nor Yasuda et al. teaches or suggests the features of the present invention wherein each input/output unit includes additional information associated with the program block data including “front input/output unit data designating an input/output unit immediately preceding each input/output unit in a sequence of the machining program, and rear input/output unit data designating an input/output unit following each input/output unit in a sequence of the machining program.”

It is submitted that the identifiers U90 and V90 illustrated in Figure 3 of Miyajima et al. do not correspond to these claim features since U and V correspond to a beginning command and an end command and do not refer to a prior or succeeding input/output unit. The elements U01, V01; U02, V02; U03, V03 in the machining commands P3-P5 in Figure 3 of the Miyajima reference represent macro commands for starting and ending machining programs 101, 102 and 103 associated therewith as custom macros. These elements do not represent any information for use in designating an input/out unit immediately preceding each input/output unit or designating an input/output unit following each input/output unit in a sequence of the machining program.

It is also submitted that the pre-execution position values which were stored in the memory 53 of Yasuda et al. do not correspond to an input/output unit. In addition, such values are not stored in the input/output units which include the program block data. Instead, this data is stored in ROM 51. Further, there is no indication that this information corresponds to a feature which would indicate a prior input/output unit or a succeeding input/output unit. Further, these features were not taught by the other prior art relied on by the Examiner. In the Yasuda reference the memory means in claim 1 at column 13, lines 28-37 relied on by the Examiner, is recited to store a pre-execution in-position value and a post-execution in-position value which is defined in this paragraph, and does not disclose or suggest front input/output date designating an input/output unit immediately proceeding each input/output unit in a sequence of the machining program and rear input/output unit data designating an input/output unit following each input/output unit in the sequence of the machining program as recited in claim 1.

In summary, for the reasons set forth above, it is submitted that the prior art does not teach the claimed storage device of claim 1. Further, as explained in detail in the specification, the use of this specific additional information in the input/output units simplifies the editing/addition, deletion or use of a branching instruction in the machining program. Therefore, it is submitted that claim 1 patentably distinguishes over the prior art.

Claim 2 depends from claim 1 and includes all of the features of that claim plus additional features which are not taught or suggested by the prior art. Therefore, it is submitted that claim 2 patentably distinguishes over the prior art.

#### Rejection of Claims 3 and 4

In item 2 on pages 9 and 10 of the Office Action the Examiner rejected claims 3 and 4 as unpatentable over the prior art. Claims 3 and 4 depend, directly or indirectly from claim 1 and include all of the features of that claim plus additional features which are not taught or suggested by the prior art. Therefore, it is submitted that claims 3 and 4 patentably distinguish over the prior art.

#### Rejection of Claims 5-8

In item 13 on pages 10-12 of the Office Action the Examiner rejected claims 5-8 under 35 U.S.C. § 103 as unpatentable over the prior art. Claims 5-8 depend, directly or indirectly from claim 1 and include all of the features of that claim plus additional features which are not taught or suggested by the prior art. Therefore, it is submitted that claims 5-8 patentably distinguish over the prior art.

#### New Claims 9-12

New independent claim 9 is directed to a numerical controller which includes

a storage device storing a plurality of input/output units, each of the input/output units including program data obtained by dividing the machining program, each of the input/output units including additional information associated with the program data, the additional information including first input/output unit data designating an input/output immediately preceding each

input/output unit in a sequence of the machining program, and second input/output unit data designating an input/output unit following each input/output unit in the sequence of the machining program; and

Therefore, it is submitted that claim 9 patentably distinguishes over the prior art.

Claims 10-12 depend, directly or indirectly from claim 9 and include all of the features of that claim plus additional features which are not taught or suggested by the prior art. Therefore, it is submitted that claims 10-12 patentably distinguish over the prior art.

Summary

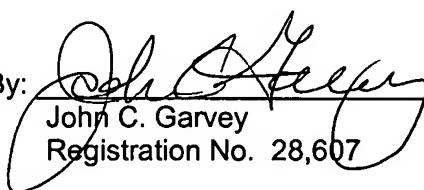
It is submitted that none of the references, either taken alone or in combination teach the present claimed invention. Thus, claims 1-12 are deemed to be in a condition suitable for allowance. Reconsideration of the claims and an early notice of allowance are earnestly solicited.

Respectfully submitted,

STAAS & HALSEY LLP

Date: 9-14-04

By:

  
John C. Garvey  
Registration No. 28,607

1201 New York Avenue, NW, Suite 700  
Washington, D.C. 20005  
Telephone: (202) 434-1500  
Facsimile: (202) 434-1501



## NUMERICAL CONTROLLER WITH REDUCED PROGRAM READING TIME

### BACKGROUND OF THE INVENTION

#### 1. Field of the Invention

**[0001]** The present invention relates to a numerical controller suitable for use in operating a machine tool while reading a machining program stored in a storage device or medium, and in particular to a numerical controller capable of efficiently executing a machining program including a branch instruction and also editing a machining program stored in the storage device or medium.

#### 2. Description of Related Art

**[0002]** In the case of DNC (direct numerical control) operation of a machine tool for operating the machine tool while reading the machining program from a storage device such as a disk drive, the machining program is sequentially read from beginning to end thereof by a unit of a character to be processed. Therefore, if a branch instruction is included in the machining program, the processing is discontinued and it is necessary to read the machining program from the beginning again for executing the branch instruction, thereby wasting a considerable amount of time. Further, for performing editing of a machining program, such as a modification, a deletion or an addition, it has been necessary to read the entire machining program to be edited, and the edited entire machining program is outputted to the storage device or medium, thereby requiring a considerable amount of time in the processing.

### SUMMARY OF THE INVENTION

**[0003]** An object of the present invention is to provide a numerical controller capable of reducing the amount of time required in operating a machine tool while reading a machining program stored in a storage device or medium, and in editing the machining program stored in the storage device or medium. Further, the present invention provides a numerical controller capable of effectively executing a machining program prepared to include a branch instruction in the storage device or medium.

**[0004]** A numerical controller in accordance with the present invention comprises: a storage device or medium for storing input/output units each including program block data obtained by dividing a machining program and additional information associated with the program block data; a processor for processing the input/output units; and an interface for inputting/outputting the input/output units between the storage device or medium and the

processor. The additional information includes an effective data length of the program block, front input/output unit data designating an input/output unit immediately preceding each input/output unit and rear input/output unit data designating an input/output unit following each input/output unit in a sequence of the machining program.

**[0005]** The processor may read a first input/output unit including a program block corresponding to a beginning part of the machining program and successively reads input/output units stored in the storage device or medium according to rear input/output unit data in the read input/output unit through the interface. The processor successively executes the program blocks included in the read input/output units.

**[0006]** When a branch instruction is included in the program block of the input/output unit in execution, the processor may read input/output units preceding the read input/output unit using the front input/output unit data and may read input/output units following the input/output unit in execution using the rear input/output unit data to search a line designated by the branch instruction.

**[0007]** The additional information may further include data specifying an input/output unit including a line designated by a branch instruction, and in this case, when the branch instruction is included in the program block of the input/output unit in execution, the processor reads the input/output unit specified by the data.

**[0008]** The processor may read only an input/output unit or input/output units to be edited from the storage device or medium through the interface.

**[0009]** In particular, the processor may read only an input/output unit to be edited and modify a program block and an effective data length included in the read input/output unit.

**[0010]** The processor may delete an input/output unit by changing rear input/output unit data of an input/output unit designated by front input/output unit data of the input/output unit to be deleted to rear input/output unit data of the input/output unit to be deleted, and changing front input/output unit data of an input/output unit designated by rear input/output data of the input/output unit to be deleted to the front input/output data of the input/output unit to be deleted.

**[0011]** The processor may add a new input/output unit including program block data and additional information. In this case, the processor changes rear input/output unit data of an input/output unit designated by front input/output data of the input/output unit to be added to data specifying the input/output unit to be added, and changes front input/output unit data

of an input/output unit designated by the rear input/output data of the input/output unit to be added to data specifying the input/output unit to be added.

**[0012]** With the above structure of the numerical controller, since the amount of data of the machining program to be inputted from the storage device or medium is reduced for executing the branch instruction and for editing the machining program, the amount of time required for input and output of the machining program is reduced.

#### BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a block diagram of a computerized numerical controller (CNC) and peripheral devices including a storage device or medium according to an embodiment of the present invention;

FIG. 2 is a diagram of a data structure of one input/output unit;

FIG. 3 is a diagram showing the relation of a first input/output unit to a fourth input/output unit;

FIG. 4 is a diagram showing input of program data from the external device to the CNC by the input/output unit;

FIG. 5 is a diagram showing output of program data from the CNC to the external device by the input/output unit;

FIG. 6 is a diagram showing editing of the program data within one input/output unit;

FIG. 7 is a diagram showing deletion of an input/output unit requiring editing of data over a plurality of input/output units; and

FIG. 8 is a diagram showing addition of an input/output unit requiring editing of data over a plurality of input/output units.

#### DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

**[0013]** A numerical controller with peripheral devices according to an embodiment of the present invention is shown in FIG. 1. As shown in FIG. 1, a computerized numerical controller (CNC) 10 for numerically controlling respective axes of a machine tool has a general hardware configuration comprising a CPU 11 for controlling a whole system, and a ROM 12, a RAM 13, a video RAM 14, a nonvolatile RAM 15, a key interface 16 and a serial interface 17 which are connected to the CPU 11 through a bus 18.

**[0014]** The video RAM 14 is connected to a CRT 19 and the nonvolatile RAM 15 is connected to a battery 20 for backing up the nonvolatile RAM 15. The key interface 16 is connected to a keyboard 21 for an operator's manual operation. The serial interface 17 is connected to a floppy disk (FD) drive 22 as a storage device for storing data using a floppy

disk as a storage medium, and a memory card interface 23 is connected to a memory card 24 as a storage medium for use in sending/receiving of programs. Details of data structure, input/output procedure and editing procedure of machining programs prepared and stored in the storage device or medium will be described later.

**[0015]** The machine tool has a general structure comprising axis controllers 31, 32 for respective axes (two axes in this embodiment), servo amplifiers 41, 42, servomotors 51, 52, as well as a programmable controller (PC) 25, a machine controller 27 and an input/output (I/O) device 26 for connecting the machine controller 27 and the PC 25.

**[0016]** In executing a machining program, data of the machining program are successively outputted from a floppy disk in the floppy disk drive 22 through the serial interface 17 or the memory card 24 through the memory card interface 23 and temporally stored in the nonvolatile RAM 15, and the CPU 11 successively reads the data of the machining program stored in the nonvolatile RAM 15 and executes the machining program to output motion commands to the axes control circuits 31, 32 for the respective axes of the machine tool.

**[0017]** The data structure of the machining program will be described referring to FIGS. 2 and 3. The machining program is divided into a plurality of program blocks and additional information is added to each of the program blocks to be associated therewith to form input/output units having the same number as that of the program blocks in one-to-one relation with the program blocks. Thus prepared input/output units are stored in the storage device or the storage medium such as the floppy disk drive 22 or the memory card 24. These input/output units are referred to as an input/output unit A, an input/output unit B, ..., an input/output unit M, an input/output unit N, ..., according to a sequence of the original machining program before dividing in this example. The number of the prepared input/output units is "n" in this example.

**[0018]** Data structure of the input/output unit M as an example of the input/output unit is shown in FIG. 2. Each input/output unit comprises a program block as a core of the input/output unit and additional information associated therewith. The program block is a part of the original machining program to be assigned to the input/output unit and comprises data of effective characters which describe statements, etc. The additional information includes front link data 61 designating an immediately preceding input/output with respect to each input/output unit in the sequence of the machining program, rear link data 62 designating a following input/output unit in the sequence of the machining program, and data 63 representing data length of the effective characters contained in the program block.

The effective character length is indicated by the number of bytes of the effective characters of the program block as represented by "sm" in this example. Counting of the effective character length is automatically carried out by an information processing device provided independently of the CNC 10 or by the CNC 10 in preparing the input/output units from the original machining program. In the case where the effective character length is increased or decreased by editing by the CNC 10, the effective character length is automatically recounted by the CNC 10.

**[0019]** Since the sequence of the machining program is such as A→B→, ..., →L→M→N, ..., contents of the front link data and the rear link data of the input/output unit M are "L" and "N", respectively. If any input/output unit is deleted or a new input/output unit is added by editing, the front link data and the rear link data may be rewritten, as described later. Further, since there is no input/output unit to be designated by the front link data in the first input/output unit A corresponding to a beginning part of the machining program, data indicating none is assigned as such front link data. Similarly, since there is no input/output unit to be designated by the rear link data in the last input/output unit corresponding to the end of the machining program, data indicating none is assigned as such rear link data.

**[0020]** In the case where a machining program includes a branch instruction, it is highly probable that a line label designated by the branch instruction is included in an input/output unit other than the input/output unit including the branch instruction. It is probable that a line of a label specified by the branch instruction exists in an input/output unit other than the input/output unit including the branch instruction. An example of four input/output units of the machining program including a branch instruction is shown in FIG. 3.

**[0021]** As shown in FIG. 3, contents of the front link data of the input/output units A, B, C and D are "NONE", "A", "B", and "C", respectively, and contents of the rear link data of the input/output units A, B, C and D are "B", "C", "D" and "E", respectively (an input/output unit E is not shown). In this example, the input/output unit C includes a branch instruction "IF [#1EQ1] GO TO N10". The meaning of "IF [#1EQ1] GO TO N10" is "go to a line designated by label "N10" if a value of the index #1 is 1". The line including the label "N10" is included in the input/output unit B. In this example of the machining program, since the value of the index #1 equals "1" only at the first time, the branch instruction is executed only once.

**[0022]** The machining program prepared in the form of input/output units and stored in the storage device or medium such as the floppy disk drive 22 or the memory card 24 is inputted into the CNC 10 for the execution or the editing of the machining program. Also, the machining program after editing in the CNC 10 may be outputted to the storage device

or medium. Procedures of the inputting and outputting of the machining program from/to the storage device or medium are shown in FIG. 4 and FIG. 5, respectively. As shown in FIGS. 4 and 5, it is not necessary to input/output data of the machining program sequentially from the beginning to the end, contrary to the conventional art. For example, the necessary input/output unit or units specified by a manual operation on the keyboard 21 may be inputted and outputted before and after the editing of the machining program.

**[0023]** Further, the machining program may be executed by the CNC 10 while reading the machining program prepared in the form of the input/output units and stored in the storage device or medium. The reading of the machining program is performed by successively reading the input/output units. The order of reading and executing the input/output units is determined according to the front/rear link data included in the additional information in each input/output unit. The input/output unit having the front link "NONE" is determined to be the first input/out unit. The CNC 10 receives data of the first input/output unit from the storage device or medium and temporarily stores the data in the internal memory, e.g. the nonvolatile RAM 15 and reads the data to start execution of the machining program.

**[0024]** Parallel to the execution of the program block in the first input/output unit, data of the second input/output unit designated by the rear link data in the first input/output unit is inputted to the CNC 10 from the storage device or medium and stored in the internal memory. The CPU 11 of the CNC 10 executes the program block in the second input/output unit after completing the program block in the first input/output unit. Subsequently, the program blocks in the third, the fourth, ..., the last input/output units are successively executed. The last input/output unit is determined by the rear link data "NONE".

**[0025]** The foregoing is a basic procedure of reading and executing the machining program. The procedure is modified if the machining program includes a branch instruction. The procedure in the case where the input/output unit includes a branch instruction will be described using the example of input/output units shown in FIG. 3. The procedure is modified in dependence on the following conditions (1) and (2).

**[0026]** (1) In the case where data (block name or number) for specifying an input/output unit including a line having a label designated by a branch instruction is prepared to be available for the CNC 10, e.g. such data is stored in the additional information of the input/output unit including the branch instruction:

**[0027]** The input/output unit including the line of the label designated by the branch instruction is read and executed irrespective of the front/rear link data of the input/output unit including the branch instruction. In the example of FIG. 3, the CNC 10 successively reads and executes the input/output unit A, the input/output unit B, the input/output unit C which includes a branch instruction, and when the branch instruction is read in the execution of a program block of the input/output unit C, the input/output unit B including the label "N10" specified by the data prepared in the additional information is read and the line of the label "N10" and subsequent lines are executed. After finishing the execution of the input/output unit B, the input/output unit C is read again and executed according to the rear link data of the input/output unit B. In the second time execution of the input/output unit C, the branch instruction is invalid since the value of #1 equals "2", and thus the procedure continues execution of the input/output unit C, the unit D, ... according to the basic rule, unless another branch instruction is read.

**[0028]** (2) In the case where the data for specifying an input/output unit including a line having a label designated by a branch instruction is not available for the CNC 10:

**[0029]** The input/output unit including the line of the label designated by the branch instruction is searched using the front/rear link data of the input/output unit including the branch instruction from peripheral input/output units.

**[0030]** In the example of FIG. 3, the CNC 10 successively reads and executes the input/output unit A, the input/output unit B, the input/output unit C which includes a branch instruction, and when the branch instruction is read in the execution of a program block of the input/output unit C, the input/output unit B is read according to the front link data of the input/output unit C. In this example, since the input/output unit B includes the line of the label "N10" designated by the branch instruction, the line of the label "N10" and subsequent lines are executed. The subsequent procedure is the same as in the case (1).

**[0031]** If the line of the label designated by the branch instruction is not included in the first read input/output unit, the input/output unit designated by the rear link data of the input/output unit including the branch instruction is read. In the example of FIG. 3, if the line of the label N10 designated by the branch instruction were not included in the first read input/output unit B, the input/output unit D is read according to the rear link data of the input/output unit C. Further, if the line of the label N10 is not included in the input/output unit D, the input/output unit A designated by the front link data of the input/output unit B is read. Thus, even if the input/output unit including the line of the label designated by the branch instruction is not directly known, there is a high probability that the line of the label

designated by the branch instruction will be located early by the search from the input/output units in the vicinity of the input/output unit including the branch instruction. Conventionally, since the whole machining program functions as a single input/output unit, the machining program is read from the beginning thereof again when the branch instruction is read, which lowers the efficiency of the procedure.

**[0032]** The editing of the machining program by the CNC 10 using the input/output units will be described with reference to FIGS. 6-8. As shown in FIGS. 4 and 5, an arbitrary one or more of the input/output units of the machining program stored in the storage device or medium can be inputted into the CNC 10 and edited on the CNC 10, and the edited input/output unit or units are stored in the storage device or medium. There are three cases in editing the machining program in practice, which can be applied to a complicated editing procedure by combination, as follows:

**[0033]** (i) Editing within one input/output unit: In this case, a part of a program block belonging to one input/output unit is edited, as shown in FIG. 6. Since this editing does not have any influence on the other input/output units and the front and rear link data, the editing procedure terminates by the editing of the program block. For instance, in the case where the editing is effected on the program data in the input/output unit M, only the input/output unit M is inputted into the CNC 10 from the storage device or medium and edited, and the edited input/output unit M is outputted to the storage device or medium to be stored. In this storage, the input/output unit before the editing is rewritten to be the edited input/output unit. The input/output unit before the editing may be preserved as historical data of the editing.

**[0034]** (ii) Editing requiring modification over a plurality of input/output units (e.g., deletion of an input/output unit): The editing in this case is to delete an input/output unit including an unnecessary program block. In addition to the deletion of the input/output unit, it is necessary to modify rear link data of an input/output unit designated by the front link data of the input/output unit to be deleted, and front link data of an input/output unit designated by the rear link data of the input/output unit to be deleted.

**[0035]** For instance, in the case where the input/output unit C is to be deleted as shown in FIG. 7, the rear link data of the input/output unit B is changed from "C" to "D", and the front link data of the input/output unit D is changed from "C" to "B". For the above editing, the input/output units B, C and D are inputted into the CNC 10 from the storage device or medium according to a command input from the keyboard 21, and the input/output units B, C and D are edited and then outputted to the storage device or medium to be stored.

therein. In this procedure, the input/output units B and D before the editing are rewritten to store the edited input/output units and the input/output unit C is deleted. The input/output units before the editing may be preserved as historical data of the editing.

**[0036]** (iii) Editing requiring modification over a plurality of input/output units (e.g., addition of an input/output unit): Editing in this case is to add a new input/output unit. As well as the addition of data of the new input/output unit, it is necessary to modify rear link data of an input/output unit designated by the front link data of the input/output unit to be added, and front link data of an input/output unit designated by the later link data of the input/output unit to be added.

**[0037]** For instance, in the case where a new input/output unit CC is to be added between the input/output unit C and the input/output unit D, as shown in FIG. 8, the rear link data of the input/output unit C is changed from "D" to "CC", and the data of the front link of the input/output unit D is changed from "C" to "CC". The front link data and the rear link data of the new input/output unit CC are set to "C" and "D", respectively. For the above editing, the input/output units C and D are inputted into the CNC 10 from the storage device or medium according to a command input from the keyboard 21, and the input/output units C and D are modified and the new input/output unit CC is added and then outputted to the storage device or medium to be stored therein. In this procedure, the input/output units C and D before the editing are rewritten to store the edited input/output units and the new input/output unit CC is added. The input/output units before the editing may be preserved as historical data of editing.

**[0038]** As described, according to the present invention, the machining program can be outputted from the storage device or medium to the CNC and vice versa in the form of input/output units, so that the sending/receiving of the data necessary for execution or editing of the machining program are performed in a shorter time in comparison with the conventional input/output of the whole machining program as a unit. Particularly in operating a machine tool while reading a machining program including branch instructions from a storage device or medium when a branch instruction is read in executing the machining program, a line designated by the branch instruction is determined at high speed. Thus, the reading of the machining program is performed at high speed to shorten a machining time in comparison with the conventional art where the execution of the machining program is suspended to await completion of reading the machining program.

**[0039]** Further, the editing of the machining program stored in the storage device or medium can be performed by inputting only the input/output unit or units to be edited from

the storage device or medium into the CNC, so that time required for editing the machining program is shortened.

## ABSTRACT OF THE DISCLOSURE

A numerical controller capable of quickly sending/receiving machining program data to/from a storage device or medium and quickly editing the machining program data. The machining program is divided into a plurality of program blocks and additional information is added to each of the program blocks to be associated therewith to form input/output units. The additional information includes front and rear link data designating input/output units immediately preceding and following each input/output unit, respectively, in a sequence of the machining program, and effective character length data of the program block. In executing the machining program while reading the machining program stored in the storage device or medium, the input/output units are successively read and executed according to the front/rear link data. When a branch instruction is included in the program block in execution, an input/output unit including a line having a label designated by the branch instruction is searched using the front/rear link data, or the line is read according to data for specifying such input/output unit prepared to be associated with the branch instruction in advance. In editing the machining program, only the input/output units to be edited are read from the storage device or medium and edited.